REMARKS

Applicant amended independent claim 1 to add a feature similar to the feature recited in old claim 9 that "a branch based on a bit specified in the branch instruction of a register specified in the branch instruction being set or cleared." Applicant notes that the specification describes this feature at page 20, line 16, to page 21, line 11, of the originally filed application. Applicant similarly amended independent claims 17 and 22. Applicant also cancelled claim 9. After these amendments, claims 1-4, 6-8, 10-14 and 17-26 are pending. Claims 1, 17, and 22 are independent.

The examiner rejected claims 1, 17, 19-22 under 35 U.S.C. 102(a) and (b) as being anticipated by U.S. Patent No. 5,724,563 to Hasegawa. The examiner also rejected claims 1-4, 6-9, 11-13 under 35 U.S.C. §103(a) as being unpatentable over Hasegawa in view of U.S. Patent No. 5,394,530 to Kitta.

The examiner also rejected claim 10 under 35 U.S.C. §103(a) as being unpatentable over Hasegawa in view of Kitta, and further in view of U.S. Patent No. 5,274,770 to Khim Yeoh et al., rejected claim 14 under 35 U.S.C. §103(a) as being unpatentable over Hasegawa in view of Kitta, and further in view of U.S. Patent No. 4,742,451 to Brucker et al. The examiner further rejected claim 18 under 35 U.S.C. §103(a) as being unpatentable over Hasegawa in view Brucker, rejected claims 24-26 under 35 U.S.C. §103(a) as being unpatentable over Hasegawa in view of U.S. Patent No. 5,923,872 to Chrysos, and rejected claim 23 under 35 U.S.C. §103(a) as being unpatentable over Hasegawa in view U.S. Patent No. 5,463,746 to Brodnax et al.

With respect to old claim 9, which recited a feature similar to a feature now appearing in amended claim 1, the examiner contended that:

> **9. As to claim 9, see branch condition bit c,v in col. 11, lines 25-32 (paragraph 24 at page 7 of the Office Action)**

Applicant respectfully disagrees.

Applicant's independent claim 1 as amended now recites "executing a branch instruction in execution of an instruction stream with a branch based on a bit specified in the branch instruction of a register specified in the branch instruction being set or cleared." Applicant's branch instruction determines whether a bit, specified in the branch instruction, of a register, also

specified in the branch instruction, is set or cleared. Based on that determination, the branch instruction causes a branching operation to be performed.

Hasegawa, on the other hand, discloses a pipeline processor that can execute predictive branch instructions (Abstract). Hasegawa further describes that the format of its predictive branch instructions includes a region 21 that stores an opcode, a region 22 for specifying a branch target address, and a region 23 for storing the number of at least one instruction which is to be executed in succession after the predictive branch instruction is given before the control flow is changed (FIG. 2 and col. 6, lines 1-6). For example, as illustrated in FIGS. 5A and 5B (and as referred to by the examiner in the January 10, 2006, Advisory Action), the value 3 stored in region 23 of the predictive branch instruction helps control when the branching operation would be performed after execution of the branch instruction. Hasegawa's predictive branch instruction, however, does not include a region for specifying a bit and/or specifying a register. Accordingly, Hasegawa neither discloses nor suggests "executing a branch instruction in execution of an instruction stream with a branch based on a bit specified in the branch instruction of a register specified in the branch instruction being set or cleared," as required by applicant's independent claim 1.

Furthermore, as applicant explained in the December 27, 2005, Amendment in Reply to Final Office Action of October 26, 2005, Hasegawa's predictive branching is performed based on the execution result 109 produced by executing an instruction on execution section 11 of Hasegawa's pipeline processor (FIG. 9, col. 11). Specifically, the execution of an instruction on execution section 11 (the executed instruction is not the predictive branch instruction) causes a number of flags on the condition code 61 of the judging section 13 to be set in accordance with the result outcome produced by the execution of the instruction (FIG. 9 and col. 11, lines 20-32). These flags include the Zero flag Z, the negative flag N, the Carryover flag C, and the overflow flag V.

Hasegawa further explains that after the execution result 109 is produced, the opcode in the region 21 of the predictive branch instruction is input from the instruction decoding section 3 to the branch condition judging section 62 (col. 11, lines 33-35). For example, if the opcode received from the predictive branching instruction is "100", branching will occur if the Z flag has been set as a result of the execution of a preceding instruction (see Table 1 at col. 11).

Hasegawa's branching decisions are based on the value of the Z, N, C and V flags of the condition code 61 unit, and not on the value of a particular bit of a particular register as specified by the branching instruction.

Therefore, Hasegawa does not disclose or suggest at least the feature of "executing a branch instruction in execution of an instruction stream with a branch based on a bit specified in the branch instruction of a register specified in the branch instruction being set or cleared," as required by applicant's independent claim 1.

Kitta describes techniques for predicting a branch target address using a branch history table (Abstract). While Kitta generally discusses branching instructions, nowhere does Kitta describe the basis upon which branch instructions are performed. Kitta, therefore, does not describe that branching is performed based on a bit specified in a branching instruction of a register, also specified in the branching instruction. Accordingly, Kitta does not disclose or suggest "executing a branch instruction in execution of an instruction stream with a branch based on a bit specified in the branch instruction of a register specified in the branch instruction being set or cleared," as required by applicant's independent claim 1.

Thus, since neither Hasegawa nor Kitta discloses or suggests, alone or in combination, the feature of "executing a branch instruction in execution of an instruction stream with a branch based on a bit specified in the branch instruction of a register specified in the branch instruction being set or cleared," applicant's independent claim 1 is therefore patentable over the cited art.

Claims 2-4, 6-8, 14 and 24 depend from independent claim 1 and are therefore patentable for at least the same reasons as independent claim 1.

Independent claims 17 and 22 recite "executing a branch instruction that causes a branch operation in an instruction stream based on a bit specified in the branch instruction of a register specified in the branch instruction being set or cleared." For reasons similar to those provided with respect to applicant's independent claim 1, at least this feature is not disclosed by the cited art. Accordingly, applicant's independent claims 17 and 22 are patentable over the cited art.

Claims 18-21 and 25 depend from independent claim 17 and are therefore patentable for at least the same reasons as independent claim 17.

Claims 23 and 26 depend from applicant's independent claim 22 and are therefore patentable for at least the same reasons as independent claim 22.

Additionally, as noted, the examiner rejected claim 3 under 35 U.S.C. §103(a) as being unpatentable over Hasegawa in view of Kitta.

Applicant's claim 3 recites "[t]he method of claim 1 wherein the branch instruction further comprises an optional token that indicates a pipeline stage that the branch operation is evaluated in."

The examiner states at page 7 of the Office Action, "21. A [*sic*] to claim 3 Hasegawa also included execution of the instruction branches, and the number of the instructions to be executed (see fig. 5A)." Contrary to the examiner's position, applicant respectfully contends that Hasegawa does not disclose applicant's feature as recited in claim 3.

As Hasegawa explains in relation to FIGS. 5A and 5B:

> **FIG. 5A shows a program code sequence including a predictive branch instruction. In FIG. 5A, I.sub.1, I.sub.2, I.sub.4, I.sub.5 and I.sub.6 denote sequential instructions; I.sub.3 denotes a predictive branch instruction "Branch after 3 to X"; and X denotes a branch target instruction of the predictive branch instruction.**
>
> **FIG. 5B shows the operation of the pipeline processor 100 in the case where the instruction fetch cycle number is 2 and a predictive branch instruction I.sub.3, in which the instruction number from a current execution instruction to a branch point is 3, is executed. In this case, A=2 and B=3.**
>
> **In FIG. 5B, IF.sub.1 and IF.sub.2 denote respective "instruction fetch" cycles; ID denotes an "instruction decode" cycle; EX denotes an "instruction execution" cycle; and WB denotes a "write back" cycle.**
>
> **In the cycle C.sub.3, the predictive branch instruction I.sub.3 is decoded by the instruction decoding section 3. As a result, a value B (=3) stored in the region 23 of the predictive branch instruction I.sub.3 is input to the down counter 31 (FIG. 3) as the initial value 105. The count value Z held in the down counter 31 is initialized to 3. The count value Z held in the down counter 31 is decremented by 1 in each of the cycles C.sub.4 to C.sub.6.**
>
> **In the cycle C.sub.4, the count value Z held in the down counter 31 becomes equal to the threshold value A (=2) set by the threshold value setting section 32 (FIG. 3). As a result, in the cycle C.sub.4, the level of the selection signal 111 becomes high and the output of the selector 5 is switched from the output 112 of the adder 9 into the output 113 of the branch target address register 6. (col. 9, lines 29-58)**

Applicant contends that Hasegawa's uses the value stored in region 23 of the predictive branch instruction to control at what point the branching will take place if the branching conditions are fulfilled. However, Hasegawa's does not disclose or suggest that the pipeline stage at which the branching operation is evaluated can be controlled, and Hasegawa does not

disclose or suggest that the branch instruction itself "further comprises an optional token that indicates a pipeline stage that the branch operation is evaluated in," as required by applicant's claim 3. Accordingly, applicant's claim 3 is patentable over the cited art.

Furthermore, as noted above, the examiner rejected claim 24-26 under 35 U.S.C. §103(a) as being unpatentable over Hasegawa in view of Chrysos. Specifically, the examiner admits that "[a]s to claims 24-26 ... Hasegawa did not specifically teach the hardware based multi threaded as claimed." (paragraph 35 on page 11 of the Office Action). The examiner, however, relies on Chrysos for disclosing the above feature, and argues that "Chrysos discloses a system including hardware contexts for simultaneously multithreaded execution (see col. 12, lines 15-18)" (paragraph 35, page 11 of the Office Action).

Applicant's claim 24 recites "[t]he method of claim 1, wherein the processor is a hardware-based multithreaded processor having multiple engines to process multiple threads and the branch instruction is part of an instruction set for the multiple engines; and executing comprises: executing the branch on one of the multiple engines." Applicant's processor is a multithreaded processor that has multiple engines that can each process multiple threads. The multithreaded processor coordinates and oversees the operation of its thread-processing engines.

In contrast, Chrysos describes an apparatus for sampling values of operands of instructions in a processor pipeline of a system that has a plurality of processing stages (Abstract). Chrysos discloses that the computer system 100, shown in FIG. 1, may include one or more processors 110 (col. 7, line 66 to col. 8, line 1), and that some of those processors may have multiple logical registers sets, thereby enabling processing of simultaneously executing threads (col. 12, lines 15-18). However, nowhere does Chrysos disclose a processor that includes a main multithreaded processor that has multiple engines to process multiple threads. Thus, Chrysos does not disclose or suggest a processor that "is a hardware-based multithreaded processor having multiple engines to process multiple threads and the branch instruction is part of an instruction set for the multiple engines; and executing comprises: executing the branch on one of the multiple engines," as required by applicant's claim 24.

Since neither Hasegawa nor Chrysos discloses or suggests, alone or in combination, at least the feature of "a hardware-based multithreaded processor having multiple engines to process multiple threads and the branch instruction is part of an instruction set for the multiple

engines; and executing comprises: executing the branch on one of the multiple engines," applicant's claim 24 is therefore patentable over the cited art.

Claims 25 and 26 recite "a hardware-based multithreaded processor having multiple engines to process multiple threads and the branch instruction is part of an instruction set for the multiple engines; and executing and deferring occurs on one of the multiple engines," or similar language. For reasons similar to those provided with respect to claim 24, at least this feature is not disclosed by the cited art. Accordingly, claims 25 and 26 are patentable over the cited art.

It is believed that all the rejections and/or objections raised by the examiner have been addressed.

All of the dependent claims are patentable for at least the reasons for which the claims on which they depend are patentable.

Canceled claims, if any, have been canceled without prejudice or disclaimer.

Any circumstance in which the applicant has (a) addressed certain comments of the examiner does not mean that the applicant concedes other comments of the examiner, (b) made arguments for the patentability of some claims does not mean that there are not other good reasons for patentability of those claims and other claims, or (c) amended or canceled a claim does not mean that the applicant concedes any of the examiner's positions with respect to that claim or other claims.

Enclosed is a Petition for One Month Extension of Time. Please apply the required fee of $120, and any other charges, to deposit account 06-1050, referencing attorney docket 10559-311US1.

Respectfully submitted,

Date: Feb. 23, 2006

Ido Rabinovitch
Attorney for Intel Corporation
Reg. No. L0080

Fish & Richardson P.C.
Telephone: (617) 542-5070
Facsimile: (617) 542-8906

21272788.doc